

AD-A185 729 PROGRESS IN MATHEMATICAL PROGRAMMING(U) IBM ALMADEN
RESEARCH CENTER SAN JOSE CA N MEGIDDO 04 MAR 87
\$N00014-87-G-0037

AD-A185 729 PROGRESS IN MATHEMATICAL PROGRAMMING(U) IBM ALMADEN
RESEARCH CENTER SAN JOSE CA N MEGIDDO 04 MAR 87
\$N00014-87-G-0037

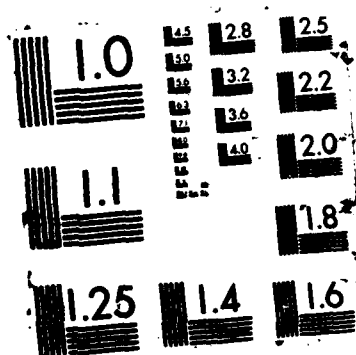
AD-A185 729 PROGRESS IN MATHEMATICAL PROGRAMMING(U) IBM ALMADEN 1/1
RESEARCH CENTER SAN JOSE CA N NEGIDDO 04 MAR 87
\$N00014-87-G-0037

UNCLASSIFIED F/G 12/4

UNCLASSIFIED F/G 12/4

UNCLASSIFIED F/G 12/4 NL

A 2x15 grid of 30 small, mostly black and white images, likely representing different types of debris or objects found at the crash site. The images are arranged in two rows of 15. The top row contains 14 images, and the bottom row contains 16 images. The images are mostly black and white, with some showing faint details of objects or debris.



REPORT DOCUMENTATION PAGE

1. REPORT SECURITY CLASSIFICATION

Non-Classified

1b. RESTRICTIVE MARKINGS

None

3. DISTRIBUTION/AVAILABILITY OF REPORT

DISTRIBUTION STATEMENT A

Approved for public release

5. MONITORING ORGANIZATION

AD-A185 729

NAME OF PERFORMING ORGANIZATION

International Business
Machines Corporation (IBM)6b. OFFICE SYMBOL
(If applicable)

7a. NAME OF MONITORING ORGANIZATION

ADDRESS (City, State and ZIP Code)

650 Harry Road
San Jose, CA 95120-6099

7b. ADDRESS (City, State and ZIP Code)

NAME OF FUNDING/SPONSORING
ORGANIZATION8b. OFFICE SYMBOL
(If applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

Grant No. N00014-87-G-0037

ADDRESS (City, State and ZIP Code)

10. SOURCE OF FUNDING NOS.

PROGRAM
ELEMENT NO.PROJECT
NO.TASK
NO.WORK UNIT
NO.

TITLE (Include Security Classification) Non-Classified

Conference on Progress in Mathematical Programming

PERSONAL AUTHOR(S)

Individual Authors on each paper on proceedings

TYPE OF REPORT

Proceedings

13b. TIME COVERED

FROM 03/01/87 TO 03/04/87

14. DATE OF REPORT (Yr., Mo., Day)

03/04/87

15. PAGE COUNT

SUPPLEMENTARY NOTATION

Proceedings on conference on Progress in Mathematical Programming

COSATI CODES

FIELD GROUP SUB. GR.

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

ABSTRACT (Continue on reverse if necessary and identify by block number)

This Proceedings is a collection of papers presented at the Conference on Progress in Mathematical Programming held at Asilomar in March of 1987. The articles are based on invited talks. The speakers are from a wide range of universities and corporations.

DTIC
ELECTE
S OCT 23 1987 D
CD D

DISTRIBUTION/AVAILABILITY OF ABSTRACT

CLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☒ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION

Non-Classified

NAME OF RESPONSIBLE INDIVIDUAL

22b. TELEPHONE NUMBER
(Include Area Code)

22c. OFFICE SYMBOL

Research Report

Report on the conference: PROGRESS IN MATHEMATICAL PROGRAMMING

Nimrod Megiddo
IBM Almaden Research Center
650 Harry Road
San Jose, CA 95120-6099 and
School of Mathematical Sciences
Tel Aviv University
Tel Aviv, Israel



Accession For	
NTIS	CRA&I <input checked="" type="checkbox"/>
DTIC	TAB <input type="checkbox"/>
Unannounced <input type="checkbox"/>	
Justification	
By <i>per ltr.</i>	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

IBM

Research Division
Yorktown Heights, New York • San Jose, California • Zurich, Switzerland

Report on the conference

PROGRESS IN MATHEMATICAL PROGRAMMING *

Asilomar Conference Center, Pacific Grove, California

March 1-4, 1987

Abstract. The main topic of the conference was developments in the theory and practice of linear programming since Karmarkar's algorithm. There were thirty presentations and about fifty people attended. Presentations included new algorithms, new analyses of algorithms, reports on computational experience, and some other topics related to the practice of mathematical programming.

* Host organization: IBM. Financial support from the Office of Naval Research and from IBM Corporation is gratefully acknowledged. Organized by Nimrod Megiddo, IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120-6099, and School of Mathematical Sciences, Tel Aviv University, Tel Aviv, Israel.

(2)

The conference "Progress in Mathematical Programming" was held at the Asilomar conference center in Pacific Grove, California, March 1-4, 1987. This conference followed a previous one, titled "New directions in Mathematical Programming", which was held at the Postgraduate Naval School in Monterey in February 1986 in Monterey, California. The idea of the second conference was to have a follow up meeting for exchanging information on progress made since the first conference.

Interestingly, most of the progress reported at the conference was on the theoretical side. Several new polynomial algorithms for linear programming were presented, (Barnes-Chopra-Jensen, Goldfarb-Mehrotra, Kojima-Mizuno-Yoshise, Renegar, Todd, Vaidya, and Ye). The common feature to most of the new polynomial algorithms is the path-following aspect. The method of McCormick-Sofer for convex programming also follows a path. Other algorithms were presented were by Betke-Gritzmann, Blum, Gill-Murray-Saunders-Wright, Nazareth, Vial, and Zikan-Cottle. Efforts in the theoretical analysis of algorithms was also reported, (Anstreicher, Bayer-Lagarias, Imai, Lagarias, Megiddo-Shub, Lagarias, Smale, and Vanderbei). Computational experiences were reported by Lustig, Tomlin, Todd, Tone, Ye, and Zikan-Cottle. Of special interest, although not in the main direction discussed at the conference, was the report by Rinaldi on the practical solution of some large traveling salesman problems. At the time of the conference it was still not clear whether the new algorithms developed since Karmarkar's algorithm would replace the simplex method in practice. Alan Hoffman presented results on conditions under which linear programming problems can be solved by greedy algorithms. In other presentations, Fourer-Gay-Kernighan presented a programming language (AMPL) for mathematical programming, David Gay presented graphic illustrations of the performance of Karmarkar's algorithm, and James Ho discussed possible embedding of linear programming in commonly used spreadsheets.

The participants in the conference were Ilan Adler, Kurt Anstreicher, Earl Barnes, David Bayer, Lenore Blum, Richard Cottle, Robert Fourer, David Gay, Philip Gill, Neal Glassman, Donald Goldfarb, Clovis Gonzaga, Harvey Greenberg, Peter Gritzmann, James K. Ho, Alan Hoffman, T. C. Hu, Hiroshi Imai, Jeffrey Kennington, Masakazu Kojima, Jeffrey C. Lagarias, Irvin Lustig, Roy Marsten, Garth McCormick, Nimrod Megiddo, Sanjay Mehrotra, Walter Murray, Larry Nazareth, Prakash Ramanan, James Renegar, Mauricio Resende, Giovanni Rinaldi, Linus Schrage, David Shanno, Mike Shetty, Michael Shub,

Steve Smale, Michael Todd, John Tomlin, Kaoru Tone, Pravin Vaidya, Robert Vanderbei, Jean-Philippe Vial, Geraldo Veiga, Yinyu Ye, and Karel Zikan.

Abstracts of the papers are given below. A volume of refereed proceedings is now in preparation. However, not all the papers presented at the meeting will be represented in this volume of proceedings. Some of those had already been submitted to journals before the conference. The interested readers should contact the authors directly for copies of their papers.

ABSTRACTS

1. **THE PROJECTIVE SUMT METHOD FOR CONVEX PROGRAMMING**, Garth P. McCormick, Department of Operations Research, SEAS, GWU, Washington, DC 20052 and Ariela Sofer, Department of Operations Research and Applied Mathematics, George Mason University, Fairfax, Virginia 22030.

The projective SUMT method is derived from the differential equation characterizing the trajectory of unconstrained minimizers of the classical logarithmic barrier function method. The continuous version requires the solution of a differential equation. The discrete version generates at each iteration the same search direction and uses as the step length the solution of the step size problem based on the logarithmic methods of centers. Convergence to a global solution of a convex programming problem can be proved under minimal assumptions. A version of the algorithm which handles linear equality constraints is similar to Karmarkar's method. A polynomial bound on the number of iterations is shown under assumptions less restrictive than those he invokes.

2. **PROJECTION ALGORITHMS FOR LINEAR PROGRAMMING**, U. Betke Universität Siegen, 3, Postf. 101240, D-5900 Siegen, W. Germany, and Peter Gritzmann Department of Mathematics, University of Washington, Seattle, WA 98195.

Based on nearest-point projection approach we generalize the relaxation methods of Agmon, Motzkin and Schoenberg to solve the feasibility problem of linear programming.

Application of Shor's method of space dilatation gives rise to a series of polynomial ellipsoidal algorithms with improved termination in case of infeasibility. Moreover, making further use of our general projection approach which renders possible applications of variable metric algorithms with exact line search, we obtain a fast and practically well-behaving algorithm for linear programming.

3. PRELIMINARY COMPUTATIONAL EXPERIENCE WITH AN INTERIOR POINT METHOD IN MPSIII, J. A. Tomlin, Ketron Management Science, Inc. Mountain View, CA 94040

Integration of an interior point algorithm with a large-scale production mathematical programming system (MPSIII) and computational experience with this system are discussed. In particular we consider comparisons with state-of-the-art simplex implementations for various classes of models.

4. GREEDY ALGORITHMS FOR LINEAR PROGRAMMING, A.J. Hoffman, IBM T.J. Watson Research Center, Yorktown Heights, New York 10598, and Department of Operations Research, Stanford University, Stanford, CA 94305

We describe some results of the last two years on the topic and advertise pertinent unsolved problems. There are two principal themes:

A) Simple combinatorial optimization problems. ONE theorem of linear programming simultaneously validates algorithms for minimum spanning tree (and minimum rooted arborescence). Dijkstra shortest path algorithm (forward, backward, "mixed"), coloring interval graphs, Frechet bounds on bivariate distribution with prescribed marginals, etc.

B) Series - parallel graphs. (1) It is shown that a nonnegative A has the property: for all b, \bar{x} maximal in $P \equiv \{x | Ax \leq b, x \geq 0\}$ implies \bar{x} maximizes $(\bar{1}, x)$ on P : if and only if A arises in a certain way from some series parallel graph. (2) Results of Aneja, Chandrasekaran, Nair, Bein, Brucker, Tamir on optimum flows in series parallel graphs are generalized in various ways. The central question here is to find properties of a function $F(,)$ of two variables such that: if $\sum x_i A_i \leq a, x \geq 0, \max \sum c_i x_i$ and $\sum y_j B_j \leq b, y \geq 0, \max$

$\sum d_j y_j$ are both solved by greedy algorithms, then so is

$$\sum z_{ij} \begin{bmatrix} A_i \\ B_j \end{bmatrix} \leq \begin{bmatrix} a \\ b \end{bmatrix}, z \geq 0, \max \sum F(c_i, d_j) z_{ij}.$$

5. **IMPROVED BOUNDS AND CONTAINING ELLIPSOIDS IN KARMARKAR'S LINEAR PROGRAMMING ALGORITHM**, Michael J. Todd, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY. Research supported in part by NSF Grant ECS-8602534 and the U.S. Army Research Office through the Mathematical Sciences Institute of Cornell University.

Karmarkar's projective algorithm for linear programming provides not only primal solutions but dual solutions giving bounds on the optimal value. Here we show how improved bounds can be obtained at the expense of solving a two-dimensional linear programming problem at every iteration, and also how an ellipsoid containing all dual optimal solutions can be generated from available information. We also give the results of limited computational experiments related to these topics.

6. **BOUNDARY BEHAVIOR OF INTERIOR POINT ALGORITHMS IN LINEAR PROGRAMMING**, Nimrod Megiddo, IBM Almaden Research Center, 650 Harry Road, San Jose, California 95120-6099, and Tel Aviv University, Tel Aviv, Israel, and Michael Shub, IBM T.J. Watson Research Center, Box 218, Yorktown Heights, New York 10598.

This paper studies the boundary behavior of some interior point algorithms for linear programming. The algorithms considered are Karmarkar's projective rescaling algorithm, the linear rescaling algorithm which was proposed as a variation on Karmarkar's algorithm, and the logarithmic barrier technique. The study includes both the continuous trajectories of the vector fields induced by these algorithms and also the discrete orbits. It is shown that, although the algorithms are defined on the interior of the feasible polyhedron, they actually determine differentiable vector fields on the closed polyhedron. Conditions are given under which a vector field gives rise to trajectories, that each visit the neighborhoods of all the vertices of the Klee-Minty cube. The linear rescaling algorithm satisfies these conditions. Thus, limits, of such trajectories obtained when a starting point is pushed to

the boundary, may have an exponential number of breakpoints. It is shown that limits of projective rescaling trajectories may have only a linear number of such breakpoints. It is however shown that projective rescaling trajectories may visit the neighborhoods of linearly many vertices. The behavior of the linear rescaling algorithm near vertices is analyzed. It is shown that all the trajectories have a unique asymptotic direction of convergence to the optimum.

7. A NEWTON'S METHOD INTERPRETATION OF KARMARKAR'S ALGORITHM FOR LINEAR PROGRAMMING, David Bayer, Box 13, Department of Mathematics, Columbia University, New York, N.Y. 10027, and J. C. Lagarias, 2C-373. AT&T Bell Laboratories, 600 Mountain Avenue, Murray Hill, N.J. 07974-2070.

Let A be an $m \times n$ matrix of full rank with entries in R , and let b, c be vectors in R^n . Let $P \subset R^n$ be the polytope defined by the system of inequalities $Ax + b \geq 0$ and let $c \cdot x$ be a linear objective function defined on P . Assume that P has a nonempty interior $\text{Int } P$, and that an interior point x_0 is known. Assume also that $c \cdot x$ attains its minimum on P at a unique optimum point $x_{opt} \in \text{Int } P$, and that the minimum value $c_{opt} = c \cdot x_{opt}$ is known. In this talk, we study Karmarkar's algorithm [Kar84] for solving the linear programming problem: "find x_{opt} minimizing $c \cdot x$, subject to $Ax + b \geq 0$." Karmarkar's algorithm is based on a vector field $\Phi_K : \text{Int } P \rightarrow R^n$, and a potential function $g : \text{Int } P \rightarrow R$ which is minimized in the limit by sequences of points approaching x_{opt} . Starting with the interior point x_0 , successive iterates x_{j+1} are computed recursively from x_j by the formula: $x_{j+1} = x_j + t\Phi_K(x_j)$, for values of $t > 0$ chosen so that the sequence $\{x_j\}$ converges to x_{opt} . We consider the version of Karmarkar's algorithm where t is chosen at each step to minimize the potential function g along the ray $x_j + t\Phi_K(x_j)$.

The derivation in [Kar84] of Karmarkar's algorithm is based on the use of projective transformations which center the successive iterates x_j in transformed polytopes P' . For this purpose, Karmarkar works with polytopes given by $m - n$ equality constraints on the first orthant of R^m . Using techniques of classical projective geometry, we explain how to directly make projective transformations of polytopes given in inequality form, and we give a second derivation of Karmarkar's algorithm in this setting.

One could instead use Newton's method to minimize g inside the polytope P , again choosing step lengths which minimize g in each step direction; this algorithm would also produce a sequence of iterates $\{x_j\}$ converging to x_{opt} . Newton's method, computed in our original coordinate system, is not the same as Karmarkar's algorithm; however, Newton's method is not invariant under projective transformations. We consider a projective transformation which maps the objective hyperplane $c \cdot x - c_{opt}$ to the hyperplane at infinity; the potential function g assumes a particularly simple form in these coordinates. We find that Newton's method computed in these coordinates agrees exactly, step by step, with Karmarkar's algorithm.

We shall only be considering the steps $x_{j+1} - x_j$ taken by iterations of Karmarkar's algorithm. For a discussion of how linear programming problems can be brought into our initial form, and of other aspects of Karmarkar's algorithm not considered in this talk, see [Kar84]. For a presentation of a related interior point algorithm for linear programming based on Newton's method, see [Ren86]. For further development of the approach to studying centers used in this talk, see [BL86].

Bibliography

- [BL86] D. Bayer, J.C. Lagarias, The non-linear geometry of linear programming, I. Affine and projective scaling trajectories, II. Legendre transform coordinates, III. Central trajectories, preprints, AT&T Bell Laboratories (1986).
- [Kar84] N. Karmarkar, A new polynomial-time algorithm for linear programming, *Combinatorica* 4 (1984), pp. 373-395.
- [Ren86] J. Renegar, A polynomial-time algorithm, based on Newton's method for linear programming, preprint MSRI 07118-86 (1986), Mathematical Sciences Research Institute, Berkeley, CA.

8. A POLYNOMIAL TIME ALGORITHM, BASED ON NEWTON'S METHOD, FOR LINEAR PROGRAMMING, James Renegar, Department of Operations Research, Stanford University, Stanford, CA 94305.

We present a new interior method for linear programming. It is conceptually simpler than Karmarkar's algorithm. Also, it has a proven worst case bound that is slightly better than Karmarkar's proven bound.

9. AN ALGORITHM FOR LINEAR PROGRAMMING WHICH REQUIRES $O(((m+n)n^2 + (m+n)^{1.5}n)L)$ ARITHMETIC OPERATIONS, Pravin M. Vaidya, AT&T Bell Laboratories, Murray Hill, N.J.

We present an algorithm for linear programming which requires $O(((m+n)n^2 + (m+n)^{1.5}n)L)$ arithmetic operations where m is the number of inequalities, and n is the number of variables. Each operation is performed to a precision of $O(L)$ bits. L is bounded by the number of bits in the input. This algorithm is faster than Karmarkar's algorithm by a factor of $\sqrt{m+n}$.

10. AMPL: A MATHEMATICAL PROGRAMMING LANGUAGE, Robert Fourer, Northwestern University, Evanston, Illinois 60201, AT&T Bell Laboratories, Murray Hill, New Jersey 07974 David M. Gay, AT&T Bell Laboratories, Murray Hill, New Jersey 07974, and Brian W. Kernighan AT&T Bell Laboratories, Murray Hill, New Jersey 07974

Practical large-scale mathematical programming involves more than just the minimization or maximization of an objective function subject to constraint equations and inequalities. Before any optimizing algorithm can be applied, some effort must be expended to formulate the underlying model and to generate the requisite computational data structures.

If algorithms could deal with optimization problems on the same terms as people, then the formulation and generation phases might be relatively easy. In reality, however, there are many differences between the form in which human modelers understand a problem and the form in which algorithms solve it. Reliable *translation* from the "modeler's form" to the "algorithms form" is often a considerable expense.

In the traditional approach to translation, the work is divided between human and computer. First, a person who understands the modeler's form writes a computer program whose output will be the required data structures. Then a computer compiles and executes the program to create the algorithm's form. This arrangement is often costly and error-prone; most seriously, the program must be debugged by a human modeler even though its output - the algorithm's form - is not meant for people to read.

In the important special case of linear programming, the largest part of the algorithm's form is the representation of the constraint coefficient matrix. Typically this is a very sparse matrix whose rows and columns number in the hundreds or thousands, and whose nonzero elements appear in intricate patterns. A computer program that produces a compact representation of the coefficients is called a matrix generator. Several programming languages have been designed specifically for writing linear programming matrix generators, and standard languages like Fortran are also often used.

Many of the difficulties of translation from modeler's form to algorithm's form can be circumvented by the use of a computer modeling language for mathematical programming. A *modeling language* is designed to express the modeler's form in a way that can serve as direct input to a computer system. Then the translation to the algorithm's form can be performed entirely by computer, without the intermediate stage of programming.

We describe in the paper this design and implementation of AMPL, a new modeling language for mathematical programming. AMPL is notable for the generality of its syntax, and for the similarity of its expressions to the algebraic notation customarily used in the modeler's form of a problem. It offers a variety of types and operations for the definition of indexing sets, as well as a range of logical expressions.

We intend AMPL to be able to express arbitrary mathematical programming problems, including ones that incorporate nonlinear expressions or discrete variables. However, our initial implementation is restricted to linear expressions in continuous variables. Thus AMPL is introduced by means of a simple linear programming example; subsequent sections examine major aspects of the language's design in more detail, with reference to three much more complex linear programs. We also discuss a standard representation of the data for a model, and describe our initial implementation of a translator that can interpret an AMPL model and associated data. Finally, we compare AMPL to the languages used by various linear programming systems, and indicate how AMPL is likely to be extended and integrated with other modeling software.

11. ON THE MULTIPLICATIVE PENALTY FUNCTION FOR LINEAR PROGRAMMING. Hiroshi IMAI. School of Computer Science. McGill University 805 Sherbrooke Street West. Montreal. PQ. Canada H3A 2K6. and Department of

Computer Science and Communication Engineering, Kyushu University, Hakozaki,
Fukuoka 812, Japan

The multiplicative penalty function method for linear programming is introduced by Iri and Imai, which is a Newton-like descent algorithm for minimizing the multiplicative penalty function, an affine analogue of Karmarkar's potential function. It is shown that the multiplicative penalty function is convex and the algorithm converges superlinearly when the optimum value of the linear objective function is given in advance.

In this talk, we will present several extensions of the multiplicative penalty function method. Specifically, we extend the multiplicative penalty function method so that it can handle the problem of unknown optimum value directly. The extended algorithm generates convergent dual solutions, where a new duality on the multiplicative function plays an important role. This duality is on interior points of both primal and dual problems, not on extreme points, and is discussed in detail. We also give a sufficient condition for a constraint to be inactive at all optimum solutions, which can be checked in the process of the algorithm. We finally mention some connections of our algorithm with Sonnevend's and Renegar's methods, and further refer to the strict convexity of the multiplicative version of Karmarkar's potential function when the corresponding feasible region is bounded.

12. A BRANCH-AND-CUT ALGORITHM FOR THE RESOLUTION OF LARGE SCALE SYMMETRIC TRAVELING SALESMAN PROBLEMS, Giovanni Rinaldi. IASI - CNR viale Manzoni 30, Roma Italy, and New York University, 40 W 4th Street Rm 524, New York, New York 10003

A branch-and-cut algorithm is presented for the resolution of large scale symmetric traveling salesman problems.

The basic idea of branch-and-cut marries linear programming based cutting planes techniques with branching techniques.

The cuts generated by the algorithm are inequalities that define facets of the TSP polytope, and that are violated by the optimal solution of a LP relaxation. The identification of violated inequalities is carried out by exact algorithms for the so-called subtour elimination

and the 2-matching constraints and by efficient heuristics for the comb and clique-trees constraints. Computational results are reported on a wide sample of test problems solved to optimality by the algorithm. The sizes of these problems ranges from 100 to more than 2000 cities.

13. PRICING CRITERIA IN LINEAR PROGRAMMING, J. L. Nazareth, Center for Pure and Applied Mathematics University of California, Berkeley, CA

We propose a reduced-gradient technique for linear programming which is motivated by the approach of Karmarkar but which builds more directly on the simplex method. We formulate a mathematical algorithm, discuss questions that arise in its implementation and describe the results of a simple yet instructive numerical experiment.

14. LINEAR PROGRAMMING WITH SPREADSHEET MACROS, James K. Ho, Management Science Program College of Business Administration, University of Tennessee Knoxville, TN 37996-0562

Linear programming (LP) models have a natural tabular format. They also arise most frequently in managerial decisions which involve other forms of quantitative analysis which increasingly employ spreadsheet software. It is therefore of interest to be able to model and solve such problems directly on spreadsheets. This paper reports experience with the design and implementation of LP as spreadsheet macros using Lotus 1-2-3. Advantages and drawbacks of this approach are discussed. Suggestions are made for future development of spreadsheet software to facilitate advanced applications in numerical computation such as LP.

15. PICTURES OF KARMARKAR'S LINEAR PROGRAMMING ALGORITHM, David M. Gay, AT&T Bell Laboratories, 600 Mountain Avenue Murray Hill, NJ 07974.

Karmarkar's linear programming algorithm handles inequality constraints by changing variables to make all constraints about equally distant; it moves in the steepest-descent direction seen by the new variables. This paper summarizes four variants of Karmarkar's

linear programming algorithm (primal affine, primal projective, dual affine, and dual projective), discusses depicting polytopes (feasible regions), and presents pictures illustrating the latter three variants. These pictures give an algorithm's eye view of the variable changes and provide visual verification of certain properties.

16. THE PROBLEM OF LOWER BOUNDS IN COMPLEXITY OF LINEAR PROGRAMMING, Steve Smale, Department of Mathematics, University of California, Berkeley, CA 94720.

Consider the problem:

Minimize $c \cdot x$ subject to $x \geq 0$ and $Ax \geq b$, $x \in R^n$ and find the solution of the dual.

Lower bounds on the speed and topology of algorithms for this problem can be expressed in terms of "computation trees". Some first answers will be given.

17. VARIANTS OF KARMARKAR'S ALGORITHM, Donald Goldfarb, Department of Industrial Engineering and Operations Research, Columbia University, New York, NY 10027, and Sanjay Mehrotra, Department of Industrial Engineering, Northwestern University, Evanston, IL 60201.

Several variants of Karmarkar's algorithm are presented, including ones which do not require (i) exact computation of the projected gradient, (ii) that this direction be in the null space of the constraint matrix, and (iii) knowledge of the optimal objective value. A variant for solving homogeneous equations over a simplex is also described.

18. APPROXIMATE PROJECTIONS IN A PROJECTIVE METHOD FOR THE LINEAR FEASIBILITY PROBLEM, Jean-Philippe Vial, Department d'Economie Commerciale et Industrielle, Universite de Geneve, 2 rue de Candolle, CH-1211 Geneve, Switzerland.

The key issue in implementing a projective method is the projection operation. In order to cut down computations several authors have suggested to use approximations instead of the projection itself. Unfortunately, using approximations may not be compatible with

the proofs of polynomial complexity. We propose several types of approximations which preserve the complexity property of the version of Karmarkar's algorithm presented by de Ghellinck and Vial. The analysis is based on a relaxation of the main convergence lemma of the Ghellinck and Vial.

19. **SHIFTED BARRIER METHODS FOR LINEAR PROGRAMMING**, Philip E. Gill, Walter Murray, Michael A. Saunders and Margaret H. Wright, Department of Operations Research, Stanford University, Stanford California 94305-4022

Powell's derivation of augmented Lagrangian functions was based on shifting the constraint boundaries in a penalty-function method, thereby permitting convergence for a finite value of the penalty parameter. The analogue for barrier-function methods is to shift the location of the singularity, and to include weights on the barrier terms. The main question we shall address in this talk is how to choose the shifts and weights to ensure convergence. It will be shown that the weights and shifts may be chosen so that the minimizer of the barrier function is bounded away from a singularity. This ensures that Newton's method converges at a quadratic rate and that the region in which quadratic convergence occurs does not shrink as the solution to the LP is approached.

We may also apply the shifted approach to the dual LP. It will be shown that this leads to an algorithm whose numerical characteristics are different than those of the primal algorithm when degeneracy is present. This is not the case for the unshifted algorithm.

20. **AN ALGORITHM FOR SOLVING LINEAR PROGRAMMING PROBLEMS IN $O(n^3 L)$ OPERATIONS**, Clovis C. Gonzaga, Department of Electrical Engineering and Computer Sciences University of California Berkeley, California On leave from COPPE-Federal University of Rio de Janeiro, CX Postal 68511, 21941 Rio de Janeiro, RJ, Brasil.

This paper describes a short-step penalty function algorithm that solves linear programming problems in no more than $O(n^{0.5} L)$ iterations. The total number of arithmetic operations is bounded by $O(n^3 L)$, carried on with the same precision as that in Karmarkar's algorithm. Each iteration updates a penalty multiplier and solves a Newton-Raphson iteration on the traditional logarithmic barrier function using approximated Hessian matrices.

The resulting sequence follows the path of optimal solutions for the penalized functions as in a predictor-corrector homotopy algorithm.

21. AN IMPLEMENTATION OF A REVISED KARMARKAR METHOD, Kaoru Tone, Graduate School for Policy Science, Saitama University, Urawa, Saitama 338, Japan.

There may exist several ways of implementing the Karmarkar's algorithm. We will show one along with preliminary numerical experiments. We deal with the standard form LP. Starting from an initial interior point, one iteration of our method consists of choice of basis (factorization of basis), optimality test, reduced gradient, conjugate gradient method and determination of next point of iterate. A combination of the reduced gradients and the conjugate gradient methods is used for generating an approximation of the steepest descent direction of the Karmarkar's potential function. Our method can deal both with the projective transformation and with the affine transformation. Bases which are maintained and updated throughout the iterations are effectively utilized. As a basis, we choose the linearly independent columns of the coefficient matrix corresponding to the decreasing order of the variables. The basis is then factorized in the LU-form which is utilized in the computations throughout the iterations. Preliminary numerical experiments on problems with dense matrices as well as with sparse ones will be reported. From the limited experiences, we know that about 50-80 percent of the CPU time is spent in the CG computations which strongly suggests the use of parallel processing in our implementation. Resolutions of high degeneracy and null variables are crucial points to be studied further.

22. A PRIMAL-DUAL INTERIOR POINT ALGORITHM FOR LINEAR PROGRAMMING, Masakazu Kojima, Department of Information Sciences, Tokyo Institute of Technology, Oh-Okayama, Meguro-ku, Tokyo 152, Japan, Shinji Mizuno and Akiko Yoshise, Department of Industrial Engineering and Management, Tokyo Institute of Technology, Meguro-ku, Tokyo 152, Japan.

This paper presents an algorithm that works simultaneously on primal and dual linear programming problems and generates a sequence of pairs of their interior feasible solutions. Along the sequence generated, the duality gap converges to zero at least linearly with global

convergence ratio $(1 - \eta/n)$; each iteration reduces the duality gap by at least η/n . Here n denotes the size of the problems and η a positive number depending on initial interior feasible solutions of the problems. The algorithm is based on an application of the classical logarithmic barrier function method to primal and dual linear programs, which has been recently proposed and studied by Megiddo.

23. A NEW SIMPLE HOMOTOPY ALGORITHM FOR LINEAR PROGRAMMING, Lenore Blum, Department of Mathematics and Computer Science, Mills College, Oakland, CA 94613, and Department of Mathematics, U.C. Berkeley, Berkeley, CA 94720.

We present a new homotopy algorithm for linear programming. Its salient features are its simple description, and that it arises naturally from mathematical considerations. Specifically, the algorithm is defined by a homotopy between the singular piecewise linear system (representing the given problem to be solved) and a non-singular linear system (incorporating all the problem data). In contrast to many homotopy algorithms whose starting points are independent of the particular problem (such as the Dantzig-Lemke Simplex algorithm), this algorithm utilizes all relevant data to start. While the algorithm is primarily of theoretical interest, preliminary computer experiments suggest that it counts typically favorable to Lemke pivots on large problems. In addition, the homotopy paths have interesting structure and reveal information about the condition of the original problem.

24. AN EXTENSION OF KARMARKAR'S ALGORITHM AND THE TRUST REGION METHOD FOR QUADRATIC PROGRAMMING, Yinyu Ye, Department of Engineering-Economic Systems, Stanford University, Stanford, CA 94305

An extension of Karmarkar's algorithm and the trust region method is developed for solving quadratic programming and linearly constrained programming. This algorithm is based on the objective augmentation and the projective transformation, followed by optimization over a trust ellipsoidal region. It creates a sequence of interior feasible points that converge to the optimal feasible solution. The algorithm is polynomial $O(Ln^5)$ if the objective is convex and quadratic, where we count $O(n^3)$ for solving a system of n linear equations. In this talk, I emphasize its implementation and computational results that suggest the

usefulness of this algorithm in practice.

25. THE AFFINE SCALING ALGORITHM AND PRIMAL DEGENERACY,
Robert J. Vanderbei, AT&T Bell Laboratories, Room 2C-115, Murray Hill, N.J.
07974.

Consider a linear program in standard form: Minimize $c^T x$ subject to $Ax = b$ and $x \geq 0$. Assuming nondegeneracy, the primal affine scaling algorithm is known to converge to the optimal solution. The proof of convergence uses duality theory. The algorithm calculates dual variables w according to the following formula: $w = (AD^2A^T)^{-1}AD^2c$, where D denotes the diagonal matrix containing the components of x . Under nondegeneracy this formula for w as a function of x has a unique continuous extension from the interior of the polytope to its boundary. Furthermore at the vertices this formula reduces to the usual formula one encounters when investigating the simplex method. When a vertex is primal degenerate there is no such continuous extension. We discuss the behavior of w in the vicinity of a primal degenerate vertex. We show that radial limits exist and give a simple and elegant formula for them. We hope this formula will be useful for proving convergence of the algorithm when there is primal degeneracy.

26. POLYNOMIAL TIME CONVERGENCE OF THE AFFINE SCALING ALGORITHM WITH CENTERING, Earl R. Barnes, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, Sunil Chopra, Statistics and Operations Research Department, New York University, New York, NY 10003, and David Jensen, Department of Applied Mathematics and Statistics, State University of New York, Stony Brook, NY 11794.

In the past three years a number of interior point methods have been introduced for solving linear programming problems. The affine scaling algorithm is conceptually the simplest of all these methods. However, it does not appear to converge in polynomial time as the others do. In fact, the existing proofs of convergence are based on some assumptions about nondegeneracy. So it is not known whether or not the affine scaling algorithm always converges. In this paper, we show how to modify the affine scaling algorithm to achieve polynomial time convergence in all situations where the set of optimal solutions is bounded.

We consider the following linear programming problem in standard form: Minimize $c^T x$, subject to $Ax = b$ and $x \geq 0$. Our algorithm starts with a feasible point x^0 and generates a sequence x^0, x^1, \dots , as follows:

Given x^k , compute an approximate solution ξ^k of the problem Maximize $\prod_{i=1}^n x_i$, subject to $Ax = b$, $c^T x \geq c^T x^k$, and $x \geq 0$. Define $D_k = \text{diag}(\xi_1^k, \dots, \xi_n^k)$ and compute $\lambda_k = (AD_k^2 A^T)^{-1} AD_k^2 c$. Take $x^{k+1} = \xi^k - RD_k^2(c - A^T \lambda_k) / \|D_k(c - A^T \lambda_k)\|$ where $0 < R < 1$ is a constant, independent of k .

When the approximate solution is taken to be $\xi^k = x^k$, the latter reduces to the affine scaling algorithm. We show that with very little additional work we can obtain an approximate solution resulting in an algorithm with the following property.

Let x^* denote a solution. Then $c^T x^{k+1} - c^T x^* \leq (1 - R/(2(n+1)))(c^T x^k - c^T x^*)$. It follows that we can solve the problem in $O(n)$ steps of the latter.

27. THE GEOMETRY OF LINEAR PROGRAMMING, Jeffrey C. Lagarias, AT&T Bell Laboratories, Murray Hill, NJ 07974.

A fundamental geometric object underlying Karmarkar's linear programming algorithm is the set of trajectories obtained by following the infinitesimal version of his algorithm. A rational change of variable, projective Legendre transform coordinates, is introduced that linearizes these trajectories. This change of variable maps the interior of the feasible solution polytope P to the interior of the dual polytope P^d which may be identified with the polar polytope P^o if $0 \in P$. The image trajectories are geodesics of a projectively invariant geometry on P^o , Hilbert geometry. This work relates Karmarkar's notion of centering to a projective duality between points inside P and hyperplanes outside P . It gives new invariants for measuring the progress of Karmarkar's algorithm, and may lead to better worst-case running time bounds for Karmarkar's algorithm.

28. LINEAR PROGRAMMING AND THE NEWTON BARRIER FLOW, Kurt M. Anstreicher, Yale School of Organization and Management, Box 1A, New Haven, CT 06520.

In this note we report a simple characteristic of linear programming central trajectories which has a surprising consequence. Specifically, we show that given a bounded polyhedral set P with nonempty interior, the logarithmic barrier function (with no objective component) induces a vector field of negative Newton directions which "flows" from the center of P , along central trajectories, to solutions of every possible linear program on P

29. THE BOX METHOD: A NEW INTERIOR-POINT ALGORITHM FOR LINEAR PROGRAMMING, Karel Zikan and Richard W. Cottle, Systems Optimization Laboratory, Department of Operations Research, Stanford University, Stanford, CA 94305.

This talk will expose a new approach to solving linear programs. Like Karmarkar's method, it generates a sequence of points belonging to the (relative) interior of the feasible region. Like the simplex method, the method is combinatorial in nature and is finite in the nondegenerate case. The method's novelty lies primarily in the simple scheme it uses to produce search directions. The underlying theory and some implementation issues of this new method will be discussed. A specialization of the algorithm to minimum cost network flow problems (Transportation Problems), and some computational experience will also be presented.

30. COMPARISONS OF COMPOSITE SIMPLEX ALGORITHMS, Irvin J. Lustig, Department of Operations Research, Stanford University, Stanford, CA 94305.

For almost forty years, the simplex method has been the method of choice for solving linear programs. The method consists of first finding a feasible solution to the problem (Phase I), followed by finding the optimum (Phase II). Many algorithms have been proposed which try to combine the processes embedded in the two-phase process. This thesis will compare the merits of some of these composite algorithms.

Theoretical and computational aspects of the Weighted Objective, Self-Dual Parametric, and Markowitz Criteria algorithms are presented. Different variants of the Self-Dual methods are discussed. A proof is presented which shows that the Self-Dual Parametric algorithm is equivalent to Lemke's algorithm when applied to linear programs.

A large amount of computational experience for each algorithm is presented. These results are used to compare the algorithms in various ways. The implementations of each algorithm are also discussed. One theme that is present throughout all of the computational experience is that there is no one algorithm which is the best algorithm for all problems.

INDEX

Author	Abstract No.
Anstreicher	28
Barnes	16
Bayer	7
Betke	2
Blum	23
Cottle	29
Fourer	10
Gay	10, 15
Gill	19
Goldfarb	17
Gonsaga	20
Gritzmann	2
Ho	14
Hoffman	4
Imai	11
Kernighan	10
Kojima	22
Lagarias	7, 27
Lustig	30
McCormick	1
Megiddo	6
Mehrotra	17
Mizuno	22
Murray	19
Nazareth	13
Renegar	8
Rinaldi	12
Saunders	19
Shub	6
Smale	16
Sofer	1
Todd	5
Tomlin	3
Tone	21
Vaidya	9
Vanderbei	25
Vial	18
Wright	19
Ye	24
Yoshise	22
Zikan	29

END

12-87

DTIC